

Excellence Through Training



**PROCEDURES FOR EDITING
WITHIN UNIX
ADP DIVISIONS**

(ALL USERS)

COMNAVAIRLANT

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP)**

FASTLANTPUB
UNXDPD - 001
REV: FEB 01

APPROVED BY:

CODE N412C

Date: _____

MEMORANDUM

From: _____
To: FASTLANT Team Project Manager

Subj: **IMPROVEMENT OF THE SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP),
RECOMMENDATIONS FOR**

1. Type of recommendation:

- | | |
|-----------------------------------|-----------------------------------|
| <input type="checkbox"/> Revision | <input type="checkbox"/> Change |
| <input type="checkbox"/> Addition | <input type="checkbox"/> Deletion |

2. The following are the recommendations for improvement of the TTP pertaining to paragraph _____:

- | | |
|-----------------------------------|--------------------------------------|
| <input type="checkbox"/> Attached | <input type="checkbox"/> As follows: |
|-----------------------------------|--------------------------------------|

(Requester's Signature)

ENDORSEMENT

From: _____
To: FASTLANT Team Project Manager

Subj: **IMPROVEMENT OF THE SUPPLY DEPARTMENT
PROFESSIONAL DEVELOPMENT PROGRAM (PDP), TASKING FOR**

1. The above recommendations are:

- | | |
|-----------------------------------|--------------------------------------|
| <input type="checkbox"/> Approved | <input type="checkbox"/> Disapproved |
|-----------------------------------|--------------------------------------|

2. Request take appropriate action.

Date: _____
Control # _____

(Approval Signature)

This page was left blank intentionally.

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM
(TTP)**

PROCEDURES FOR EDITING WITHIN UNIX

Specific Table of Contents

SECTION 1: STUDY OUTLINE

This section provides an outline of the procedures for editing files within the UNIX system.

SECTION 2: STUDY GUIDE

This section contains a general study guide that provides detailed information for the items outlined in Section 1. Its function is to provide guidelines on how to effectively use editing procedures within UNIX.

SECTION 3: SKILLS CERTIFICATION

This section contains a questionnaire that will ensure you have the basic operating skills required to use the visual editor. It will require certification by your immediate supervisor once you complete it.

SECTION 4: HANDS-ON SKILL DEVELOPMENT

This section shows highlights of occupational skills, and a practical application of established procedures for proper use of the visual editor.

SECTION 5: TYCOM SEMINARS AND WORKSHOPS

This section contains a list of the MTAT seminars and workshops that we recommend you attend before accepting a new position in this occupational area.

SECTION 6: FUNCTIONAL DESK GUIDE

This section contains the MTAT desk guide that provides comprehensive information and detailed procedures that will help you successfully understand editing within the UNIX environment.

SECTION 7: LESSON PLAN

This section contains a MTAT lesson plan that covers basic concepts of various computer operating areas.

Excellence Through Training



**PROCEDURES FOR EDITING
WITHIN UNIX
ADP DIVISION**

**SECTION 1:
STUDY OUTLINE**

COMNAVAIRLANT

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP)**

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM
(TTP)**

PROCEDURES FOR EDITING WITHIN UNIX

SECTION 1: STUDY OUTLINE

Part A. INTRODUCTION

1. General
2. Source of Information

Part B. PROCEDURES

1. Visual Editor Description
2. Visual Editor Initiation and Termination
3. Command Mode Operations
4. Input Mode Operations
5. Last Line Mode
6. Copying Blocks of Text
7. Moving Blocks of Text
8. Writing to a Buffer Section

This page was left blank intentionally.

Excellence Through Training



**PROCEDURES FOR EDITING
WITHIN UNIX
ADP DIVISION**

**SECTION 2:
STUDY GUIDE**

COMNAVAILRANT

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP)**

PROCEDURES FOR EDITING WITHIN UNIX**STUDY GUIDE****SECTION 2****CONTROL RECORD**

Trainee Name: _____

Start Date: _____

Target Completion Date: _____

Actual Completion Date: _____

Certified By:

Div. LCPO/

Supervisor

Date

Div. Officer

Date

This page was left blank intentionally.

SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM
(TTP)

PROCEDURES FOR EDITING WITHIN UNIX

SECTION 2: STUDY GUIDE

A. INTRODUCTION

- 1. General.** This study guide covers the UNIX Visual Editor (vi) which is provided with the TAC-3 System. Although there are other editors in use at this time, we will only cover the Visual Editor. Careful monitoring and attention to detail is essential.
- 2. Source of Information.** The following is the principal source for policy and procedures for the Visual Editor (vi):
 - a. NAVMASSO's Introduction to UNIX Manual, Lesson 11;
 - b. UNIX Made Easy, Basic Editing with the Visual Editor, Chapter 2 and Advance Visual Editing, Chapter 9.

B. PROCEDURES

1. **Visual Editor Description.** The UNIX visual editor is a powerful command-driven screen editor that is included as part of the UNIX Operating System. The visual editor is used to create, modify and rearrange files. Instructions are given to the visual editor by entering a combination of keystrokes.
2. **Visual Editor Initiation and Termination.** When the vi editor is invoked, the screen is filled with the text file. The bottom line is blank for special instructions or commands. If a small file is edited the rest of the screen will display the (~) tilde in each line, which marks the unused line in the vi text display.
 - a. **Modes.** The vi editor has three modes of operation: input, command, and last line. When you first enter vi, you are in the input command mode. The command mode allows you to navigate around the screen or change the contents of the buffer. The input mode allows you to enter text into the file. The last line mode provides you with many of the advanced features of vi (importing and exporting files, or block moves of text). To invoke vi enter the following:

vi [filename] [return]

- b. **Filename.** The filename may be the name of a new file or an existing file. The new file will be created as a temporary file called buffer, in which the new text will be stored. If an existing file is being modified, the name of the file appears at the bottom left portion of the screen including the number of lines and characters in the file. If you entered the wrong filename, vi searches for that file and since it does not exist vi will create a new file, using the misspelled filename. To correct this you must enter the following:

:edit! [correct filename] [return]

- c. **Quit.** To quit vi without writing any changes or additions use the following commands:

(1) Press the escape [Esc] key.

(2) :q! [return]

- d. **Write and Quit.** To write the file and quit vi use the following option:

(1) Type :wq [Return]

📌 **NOTE:** “WARNING” <shift-zz> writes when exiting as opposed to :q!

3. Command Mode Operations.

- a. **Editing and Modifying Files.** Moving the cursor around is accomplished in the command mode by using the h, j, k, and l keys on all terminals. Each directional key can be augmented by entering a number key before it. The editor remains in the command mode when cursor movement commands are entered. The following commands apply:

j	Move cursor down one line.
k	Move cursor up one line.
h	Move cursor to the left one character.
l	Move cursor to the right one character.
0	Move cursor to the beginning of the current line.
\$	Move cursor to the end of the line.

- b. **Displaying Text.** At times you may have a file which contains more than one screen of text. To display certain portions of the text, use any of the following commands:

(1)	[Ctrl-f]	Moves forward a full screen.
(2)	[Ctrl-b]	Moves back one full screen.
(3)	[Ctrl-d]	Moves forward one-half screen.
(4)	[Ctrl-u]	Moves back a half screen.
(5)	[Ctrl-l]	Refreshes the screen.
(6)	[Ctrl-g]	Displays the current position line number.
(7)	G	Moves to the end of the file.
(8)	1G	Moves to the beginning of the file.
(9)	nG	Moves to “n” line number of the file.

4. Input Mode Operations.

- a. **Adding Text.** To add text to a file, position the cursor where you want to add the text and use one of the following commands:

(1)	i	Inserts to the left of the cursor.
(2)	a	Appends to the right of the cursor.
(3)	O	Opens a line above the cursor.
(4)	o	Opens a line below the cursor.

To exit the input mode, use the [esc] key.

- b. **Changing Text.** The change command (“C”) will put a “\$” on the last character of the line you are replacing. If more characters are entered than the current word has, vi will go into an insert mode and insert the new characters without overwriting the rest of the characters in the line. Once you finish changing the text, press the [Esc] key, which will remove the “\$” and put you back in a command mode. Changed or deleted text is placed in a buffer. To undo these changes or deletions use the following (u, U) commands:

- (1) **u** Undo the last change or deletion.
- (2) **U** Undo all changes or deletions.

📌**NOTE:** This command will undo all changes or deletions for that particular line, if you move away from that line and return to it, U will not work. These commands put you in an append mode. You must use the escape [Esc] key to get out of the append mode and back to the command mode (e.g., if you want to add and delete a character you would use the “a” to add). Once you are ready to delete a character or word, you must first use the escape key then enter the delete command.

- c. **Modify Text.** To modify text in a file, the vi editor provides several commands:

- (1) **rp** Replaces the character that the cursor is on with a “p”.
- (2) **Rstring** Will overwrite characters until the [Esc] key is depressed.
- (3) **cwstring** Overwrite the current word with the specified string.
- (4) **cctext** Replace current line with text.
- (5) **c\$text** Replace current line with text from current character to end of line.

- d. **Delete Text.** To delete text, position the cursor under the character, word or line you want to delete and use one of the following:

- (1) **x** Deletes a character.
- (2) **dw** Deletes a word.
- (3) **dd** Deletes a line.
- (4) **D** Deletes remainder of line from the position of the cursor.

For example: to delete a set of lines you can type “8dd”, which will delete lines 1 through 8 or regardless of the location of the cursor, you may type “:1,8d”.

Deleted text is stored in buffers. When a text block is deleted, the deleted text is placed into buffer 1 and the text previously deleted is shifted to buffer 2 and so forth. The shifting of deleted text from buffer to buffer continues up to buffer 9. The most recent text deletion is always stored in buffer 1. To retrieve previously deleted text use the double quote (“”). For example, in the command “3p” the double quote instructs the editor to access a buffer, the “3” indicates which buffer to retrieve and the “p” is the put command. The “p” can be upper or lower case.

5. Last Line Mode.

- a. **Last Line Mode.** The last line mode provides you with the capability to tailor or customize the vi environment. These commands are invoked by using the last line mode, colon (":") and the "set" command. Refer to the following:

- | | |
|---------------------------|--|
| (1) set all | Display all available options. |
| (2) set | Display options that have been set. |
| (3) set showmode | Display "INPUT MODE". |
| (4) set noshowmode | Turns the showmode option off. |
| (5) set number | Display line numbers while using vi. |
| (6) set nonumber | Turns line numbering off. |
| (7) set ignorecase | Ignore case during searches. |
| (8) set list | Display special characters. |
| (9) set showmatch | Match closing and opening parentheses or brackets. |
| (10) set window=n | Setting the size of a window, where the "n" enter a number without spaces and press the enter key, next enter "z", (this will redraw the window to the desired size). Keep in mind the commands "Ctrl-f" and "Ctrl-b" are affected by the "z". |

- (11) **set tabstop=n** Allows you to set the TAB to a desired position. (At the “n” enter the number of positions to set the tabs). Keep in mind that setting tab stops only affects the visual display of the text while you are in vi. If printing, the printer will set its own tabs.
- (12) **set wrapmargin=n** Allows you to set the right margin (Once this limit has been met, the line will automatically wrap to a new line. At the “n” enter the position to wrap to a new line).

b. **Cut and Paste.** To cut and paste (copy) text, vi provides some very useful functions. Ensure that you are in the command mode to use these functions. Move the cursor to the desired line of text you want copied, or if more than one line is to be copied, use the number of lines you desire to copy from the cursor position (ensure the line number precedes the yank function). Refer to the following:

- (1) **yy** This function will yank (copy) the line of text and hold it in memory.
- (2) **p** The copy in memory is put below the cursor location.
- (3) **P** The copy in memory is put above the cursor location.

📌 **NOTE:** The “p” (or “P”) function allows you to put the yanked line of text in several places. The “put” function can also be utilized with the dd commands in the same manner.

- 6. **Copying Blocks of Text.** From the vi command mode, type “:n copy n”. The first “n” is the line number to copy, the second “n” is the target line number. The copied text is placed after the target line. To reverse the copy, use the undo command (“u”). You are not limited to copying one line of text; numerous lines can be copied to a specific location in a single operation. For example, “:1,4 copy 7” will place a copy of lines 1 through 4 after line 7, while also keeping lines 1 through 4 in their original place. To set the line numbers on your screen, use the “:setnu” command.
- 7. **Moving Blocks of Text.** The move command (“m”) moves text from one location to another. If you type “:1,9 m 18”, text residing in lines 1 through 9 is moved to line 18.

8. **Writing to a Buffer Section.** To overwrite or replace an existing file, use the write command (“w”). For example, “:1,7 w file1” will overwrite file1 with the information in lines 1 through 7, only if they exist. If the file does not exist, type “:1,7 w! file1”. To append to an existing file, type “:1,7 w >> file2”.

📎NOTE: Depending on how your account is set up, the shell may not allow you to use this command.

This page was left blank intentionally.

Excellence Through Training



**PROCEDURES FOR EDITING
WITHIN UNIX
ADP DIVISION**

**SECTION 3:
SKILLS CERTIFICATION**

COMNAVAILRANT

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP)**

SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM
(TTP)

PROCEDURES FOR EDITING WITHIN UNIX

SECTION 3: SKILLS CERTIFICATION

A. INTRODUCTION

- 1. General.** Answer all questions in this section, either orally or in writing, in the presence of your immediate supervisor who will attest to your qualification based on the accuracy of your answers and your demonstrated knowledge concerning the subject matter. If you fail to qualify during this skills certification period, obtain additional training until you become fully qualified.
- 2. Certification.** Both your immediate supervisor and the Divisional Leading Chief Petty Officer or the Division Officer must note the date and initial the following pages next to every question to certify your qualification in each area.

B. QUESTIONS

	Supervisor	Certified By:		
		Date	Div. LCPO/ Div. Officer	Date
1. Which command invokes the Visual Editor?	_____	_____	_____	_____
2. Which keys are used to navigate around the Visual Editor?	_____	_____	_____	_____
3. Which key is used to exit the input mode?	_____	_____	_____	_____
4. Which command is used to display line numbers while using the Visual Editor?	_____	_____	_____	_____
5. Which command is used to overwrite a word with a specified string while using vi?	_____	_____	_____	_____
6. Which command is used to move forward one full screen?	_____	_____	_____	_____
7. Which command is used to move forward one-half screen?	_____	_____	_____	_____
8. Which command is used to display all available options?	_____	_____	_____	_____
9. Which commands are used to undo the last change?	_____	_____	_____	_____
10. What are the three modes of operation that vi has?	_____	_____	_____	_____
11. Which commands are used to change text within a file while using vi?	_____	_____	_____	_____

	Supervisor	Date	Certified By:	
			Div. LCPO/ Div. Officer	Date
12. Which commands are used to copy text?	_____	_____	_____	_____
13. Which commands are used to move text?	_____	_____	_____	_____
14. Which command is used to write and quit in vi?	_____	_____	_____	_____

This page was left blank intentionally.

C. ANSWERS

1. vi
2. h, j, k, l
3. esc
4. :set number
5. :cw(string)
6. Ctrl_f
7. Ctrl_d
8. set all
9. U
10. input, command, last line
11. replace, change
12. yank, put; and copy
13. delete, put; and move
14. :wq

This page was left blank intentionally.

Excellence Through Training



**PROCEDURES FOR EDITING
WITHIN UNIX
ADP DIVISION**

**SECTION 4:
HANDS-ON SKILLS
DEVELOPMENT**

COMNAVIAIRLANT

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP)**

SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM
(TTP)

PROCEDURES FOR EDITING WITHIN UNIX

SECTION 4: HANDS-ON SKILLS DEVELOPMENT

A. INTRODUCTION

1. General. You must complete this section (mandatory for all trainees) to receive certification as fully qualified to accomplish the specific occupational functions discussed in this package. Complete all actions discussed within this section in writing, orally or by actual demonstration. The monitoring official must ensure that you are indeed functionally qualified.

2. References. The following are the references and sources we recommend you use when you require additional information for the Visual Editor (vi):

- a. NAVMASSO's Introduction to UNIX Manual, Lesson 11;
- b. UNIX Made Easy, Basic Editing with Visual Editor, Chapter 2 and Advance Visual Editing, Chapter 9.

B. OCCUPATIONAL SKILL REQUIREMENTS

	Supervisor	Certified By:		Date
		Date	Div. LCPO/ Div. Officer	
1. Discuss the procedures for submitting trouble reports and change proposals pertaining to software problems.	_____	_____	_____	_____
2. Explain the procedures for initiating the Visual Editor.	_____	_____	_____	_____
3. Explain how to terminate the Visual Editor.	_____	_____	_____	_____
4. Discuss the procedures for creating and editing files.	_____	_____	_____	_____
5. Explain how it is possible to manipulate text.	_____	_____	_____	_____
6. Explain how you would copy text and create another file.	_____	_____	_____	_____

Excellence Through Training



**PROCEDURES FOR EDITING
WITHIN UNIX
ADP DIVISION**

**SECTION 5:
TYCOM SEMINARS
AND WORKSHOPS**

COMNAVAILRANT

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP)**

SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM
(TTP)

PROCEDURES FOR EDITING WITHIN UNIX

SECTION 5: TYCOM SEMINARS AND WORKSHOPS

- 1. Introduction.** A key element in your progress for qualifying in the UNIX Visual Editing area is your attendance at seminars and workshops sponsored by the TYCOM. This type of formal training is usually provided by the CNAL Management Training and Assistance Team (MTAT) in Building V-88 on the Naval Air Station. The MTAT provides a Seminar and Workshop Schedule to all activities annually through regular distribution channels and in the SUADPS Update Newsletter.
- 2. Recommended Seminars and Workshops.** Currently under development.

This page was left blank intentionally.

Excellence Through Training



**PROCEDURES FOR EDITING
WITHIN UNIX
ADP DIVISION**

**SECTION 6:
FUNCTIONAL DESK GUIDE**

COMNAVAILANT

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP)**

SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM
(TTP)

PROCEDURES FOR EDITING WITHIN UNIX

SECTION 6: FUNCTIONAL DESK GUIDE

1. Introduction. Attached to this cover is the MTAT desk guide that provides comprehensive information and detailed procedures that will help you operate in your new position as a Computer Operator. This desk guide is the following: Processing Within the Visual Editor (vi) FG-A5.6. After you successfully complete your studies and earn full qualification, you may begin to operate in this position. To help you continue in a successful mode should you enter new areas or encounter problems with which you are unfamiliar, this desk guide will be very handy.

This page was left blank intentionally.

Excellence Through Training



PROCESSING WITHIN THE VISUAL EDITOR (vi)

FUNCTIONAL DESK GUIDE FG-A5.6

COMNAVAIRLANT

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP)**

FASTLANTPUB
UNXDDG-001
REV: FEB 01

PROCESSING WITHIN THE VISUAL EDITOR (VI)
FUNCTIONAL DESK GUIDE
FG-A5.6

Table of Contents

<u>Section</u>	<u>Title</u>	<u>Page</u>
A	INTRODUCTION	
	1. General	1
B	RESPONSIBILITIES	
	1. ADP Officer	2
	2. Shift Supervisor	2
	3. Computer Operator	2
C	PROCEDURES	
	1. Visual Editor Initiation and Termination	3
	2. Command Mode Operations	4
	3. Input Mode Operations	4
	4. Last Line Mode	6
	5. Copying Block of Text	7
	6. Moving Blocks of Text	7
	7. Writing to a Buffer Section	7
D	RECOMMENDATIONS	
	1. General	8
	2. Lessons Learned	8
	3. References	8
E	SMA INTEREST ITEMS	
	1. General	9
F	CHECKOFF LIST	
	1. General	10

This page was left blank intentionally.

PROCESSING WITHIN THE VISUAL EDITOR (vi)

A. INTRODUCTION

1. General. The Visual Editor (vi) is provided with the implementation of the TAC-3 UNIX Operating System. The editor is used to create files and modify or arrange data within a file. It is a command-driven software, where the commands are given by entering a combination of keystrokes. The Visual Editor has different ways of accessing, moving through and making changes to files.

B. RESPONSIBILITIES

- 1. ADP Officer.** This individual is responsible for the overall operation of the ADP Division. In addition, the ADP Officer is responsible for assigning a Shift Supervisor and a Computer Operator.
- 2. Shift Supervisor.** This individual is responsible for the compliance with and proper operation of the TAC-3 UNIX Computer System. The Shift Supervisor provides guidance and direction to the Computer Operator and acts as a communications channel between user departments and the ADP Division's chain of command.
- 3. Computer Operator.** This individual is responsible for various computer operations. For the purpose of this PDP we will cover responsibilities for the Visual Editor.

C. PROCEDURES

1. Visual Editor Initiation and Termination. When the vi is invoked, the screen is filled with the text file. The bottom line is blank for special instructions or commands. If a small file is edited the rest of the screen will display the (~) tilde in each line, which marks the unused line in the vi text display.

- a. Modes.** The vi editor has three modes of operation: input, command and last line. When you first enter vi, you are in the input command mode. The command mode allows you to navigate around the screen or change the contents of the buffer. The input mode allows you to enter text into the file. The last line mode provides you with many of the advanced features of vi (importing and exporting files, or block moves of text). To invoke vi enter the following:

vi [filename] [return]

- b. Filename.** The filename may be the name of a new file or an existing file. The new file will be created as a temporary file called buffer, in which the new text will be stored. If an existing file is being modified, the name of the file appears at the bottom left portion of the screen including the number of lines and characters in the file. If you entered the wrong filename, vi searches for that file and since it does not exist vi will create a new file, using the misspelled filename. To correct this you must enter the following:


:edit! [correct filename] [return]

- c. Quit.** To quit vi without writing any changes or additions use the following commands:

- (1) Press the escape [Esc] key.
- (2) **:q! [return]**

- d. Write and Quit.** To write the file and quit vi use the following option:

- (1) Type **:wq [Return]**

 NOTE: “WARNING” <shift-zz> writes when exiting as opposed to :q!

2. Command Mode Operations.

- a. **Editing and Modifying Files.** Moving the cursor around is accomplished in the command mode by using the h, j, k, and l keys on all terminals. Each directional key can be augmented by entering a number key before it. The editor remains in the command mode when cursor movement commands are entered. The following commands apply:

- j** Move cursor down one line.
- k** Move cursor up one line.
- h** Move cursor to the left one character.
- l** Move cursor to the right one character.
- 0** Move cursor to the beginning of the current line.
- \$** Move cursor to the end of the line.

- b. **Displaying Text.** At times you may have a file which contains more than one screen of text. To display certain portions of the text, use any of the following commands:

- (1) **[Ctrl-f]** Moves forward a full screen.
- (2) **[Ctrl-b]** Moves back one full screen.
- (3) **[Ctrl-d]** Moves forward one-half screen.
- (4) **[Ctrl-u]** Moves back a half screen.
- (5) **[Ctrl-l]** Refreshes the screen.
- (6) **[Ctrl-g]** Displays the current position line number.
- (7) **G** Moves to the end of the file.
- (8) **1G** Moves to the beginning of the file.
- (9) **nG** Moves to “n” line number of the file.

3. Input Mode Operations.


- a. **Adding Text.** To add text to a file, position the cursor where you want to add the text and use one of the following commands:

- (1) **i** Inserts to the left of the cursor.
- (2) **a** Appends to the right of the cursor.
- (3) **O** Opens a line above the cursor.
- (4) **o** Opens a line below the cursor.

To exit the input mode, use the [esc] key.

- b. **Changing Text.** The change command (“C”) will put a “\$” on the last character of the line you are replacing. If more characters are entered than the current word has, vi will go into an insert mode and insert the new characters without overwriting the rest of the characters in the line. Once you finish changing the text, press the [Esc] key, which will remove the “\$” and put you back in a command mode. Changed or deleted text is placed in a buffer. To undo these changes or deletions use the following (u, U) commands:

- (1) **u** Undo the last change or deletion.
- (2) **U** Undo all changes or deletions.

 **NOTE:** This command will undo all changes or deletions for that particular line, if you move away from that line and return to it, U will not work. These commands put you in an append mode. You must use the escape [Esc] key to get out of the append mode and back to the command mode (e.g., if you want to add and delete a character you would use the “a” to add). Once you are ready to delete a character or word, you must first use the escape key then enter the delete command.

- c. **Modify Text.** To modify text in a file, vi provides several commands. Keep in mind that most of these commands puts you in the input mode:
- (1) **rp** Replaces the character that the cursor is on with a “p”.
 - (2) **Rstring** Will overwrite characters until the [Esc] key is depressed.
 - (3) **cwstring** Overwrite the current word with the specified string.
 - (4) **cctext** Replace current line with text.
 - (5) **c\$text** Replace current line with text from current character to end of line.
- d. **Delete Text.** To delete text, position the cursor under the character, word or line you want to delete and use one of the following:
- (1) **x** Deletes a character.
 - (2) **dw** Deletes a word.
 - (3) **dd** Deletes a line.
 - (4) **D** Deletes remainder of line from the position of the cursor.

For example: to delete a set of lines you can type “8dd”, which will delete lines 1 through 8 or regardless of the location of the cursor, you may type “:1,8d”.

Deleted text is stored in buffers. When a text block is deleted, the deleted text is placed into buffer 1 and the text previously deleted is shifted to buffer 2 and so forth. The shifting of deleted text from buffer to buffer continues up to buffer 9. The most recent text deletion is always stored in buffer 1. To retrieve previously deleted text use the double quote (“ ”). For example, in the command “3p” the double quote instructs the editor to access a buffer, the “3” indicates which buffer to retrieve and the “p” is the put command. The “p” can be upper or lower case.

4. Last Line Mode.


- a. **Last Line Mode.** The last line mode provides you with the capability to tailor or customize the vi environment. These series of commands are invoked by using the last line mode, colon (“:”), and the “set” command. Refer to the following:

- | | |
|---------------------------|---|
| (1) set all | Display all available options. |
| (2) set | Display options that have been set. |
| (3) set showmode | Display “INPUT MODE”. |
| (4) set noshowmode | Turns the showmode option off. |
| (5) set number | Display line numbers while using vi. |
| (6) set nonumber | Turns line numbering off. |
| (7) set ignorecase | Ignore case during searches. |
| (8) set list | Display special characters. |
| (9) set showmatch | Match closing and opening parentheses or brackets. |
| (10) set window=n | Setting the size of a window, where the “n” enter a number without spaces and press the enter key, next enter “z”, (this will redraw the window to the desired size). Keep in mind the commands “Ctrl-f” and “Ctrl-b” are affected by the “z”. |
| (11) set tabstop=n | Allows you to set the TAB to a desired position. (At the “n” enter the number of positions to set the tabs). Keep in mind that setting tab stops only affects the visual display of the text while you are in vi. If printing, the printer will set its own tabs. |

- (12) **set wrapmargin=n** Allows you to set the right margin (Once this limit has been met, the line will automatically wrap to a new line. At the “n” enter the position to wrap to a new line).

- b. **Cut and Paste.** To cut and paste (copy) text, vi provides some very useful functions. Ensure that you are in the command mode to use these functions. Move the cursor to the desired line of text you want copied, or if more than one line is to be copied, use the number of lines you desire to copy from the cursor position (ensure the line number precedes the yank function). Refer to the following:


- (1) **yy** This function will yank (copy) the line of text and hold it in memory.
- (2) **p** The copy in memory is put below the cursor location.
- (3) **P** The copy in memory is put above the cursor location.

 **NOTE:** The “p” (or “P”) function allows you to put the yanked line of text in several places. The “put” function can also be utilized with the dd commands in the same manner.

5. Copying Blocks of Text. From the vi command mode, type “:n copy n”. The first “n” is the line number to copy, the second “n” is the target line number. The copied text is placed after the target line. To reverse the copy, use the undo command (“u”). You are not limited to copying one line of text; numerous lines can be copied to a specific location in a single operation. For example, “:1,4 copy 7” will place a copy of lines 1 through 4 is after line 7, while also keeping lines 1 through 4 in their original place. To set the line numbers on your screen, use the “:setnu” command.

6. Moving Blocks of Text. The move command (“m”) moves text from one location to another. If you type “:1,9 m 18”, text residing in lines 1 through 9 is moved to line 18.

7. Writing to a Buffer Section. To overwrite or replace an existing file, use the write command (“w”). For example, “:1,7 w file1” will overwrite file1 with the information in lines 1 through 7, only if they exist. If the file does not exist, type “:1,7 w! file1”. To append to an existing file, type “:1,7 w >> file2”.

 **NOTE:** Depending on how your account is set up, the shell may not allow you to use this command.

D. RECOMMENDATIONS

1. **General.** This section provides detailed suggestions for proper processing that comes from the lessons we have learned in the past. It also provides a list of the publications we consider of most importance in this particular area of expertise.

2. **Lessons Learned.** The following is a list of the actions we recommend you execute as a part of routine business.

- a. Ensure that you understand visual editor initiation and termination procedures.
- b. Ensure that you understand requirements for editing and modifying files.
- c. Ensure that you understand how to copy and move blocks of text.

3. **References.** The following are the references and sources we recommend you use when you require additional information:

- a. NAVMASSO's Introduction to UNIX Manual
- b. UNIX Made Easy, Basic Editing with the Visual Editor

E. SMA INTEREST ITEMS

1. General. This section provides details of the most common discrepancies found during a Supply Management Assessment (SMA). Refer to this section often to ensure you maintain your standards up to or better than those prescribed by your TYCOM.

- a.** Are visual editor initiation and termination procedures being followed correctly?
- b.** Are editing and modifying requirements being performed properly?
- c.** Are copying and moving procedures properly executed for text blocking/moving?

F. CHECKOFF LIST

1. General. This section describes the separate steps necessary to complete visual editing. These steps are in the sequence that will help ensure successful completion of editing within UNIX:

- a. Review visual editor initiation and termination procedures; modes, filenames, write and quit procedures.
- b. Review editing and modifying requirements; displaying text, adding text, modifying text, changing text, deleting text, last line mode process, cutting and pasting, copying blocks of text and moving blocks of text.

Excellence Through Training



**PROCEDURES FOR EDITING
WITHIN UNIX
ADP DIVISION**

**SECTION 7:
LESSON PLANS**

COMNAVAILANT

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP)**

SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM
(TTP)

PROCEDURES FOR EDITING WITHIN UNIX

SECTION 7: LESSON PLAN

- 1. Introduction.** Attached to this cover sheet is the MTAT lesson plan that will allow you to train other personnel in the requirements and demands of your position. This lesson plan is the following: Visual Editor vi (IV-A.6). After you successfully complete your studies and earn full qualification in visual editor, you may begin to train other personnel concerning procedures and processing in this area.

This page was left blank intentionally.

Excellence Through Training



VISUAL EDITOR (vi) LESSON PLAN II-A.6

(Classroom Time 15 Minutes)

COMNAVIAIRLANT

**SUPPLY DEPARTMENT
TYCOM TRAINING PROGRAM (TTP)**

FASTLANTPUB
UNXDLP - 001
REV: FEB 01

A. INTRODUCTION.

- **General.** The Visual Editor (vi) allows the user to create, modify and arrange data within a file.
- **Generation Facility Initiation.**
 - Login utilizing the snapi logon.
 - From the menu select “xterm”.
 - Enter the command to start vi by typing in the following:

vi [filename]

Depending on the actual task required, perform file manipulation, using the procedures provided in the Desk Guide.

This page was left blank intentionally.